University of Virginia
Charles L. Brown Department of Electrical and Computer Engineering

Project SRAM – Project Proposal

Report to PICo Review Board by
Robert Costanzo, Mike Recachinas, Hector Soto

SUBJECT: This document seeks to provide a proposed solution to meet the design requirements and specifications of the high-speed 64kb cache requested by the Portable Instruments Company (PICo). The proposal will describe our approach, the novelty of our design, and the expected outcomes.

Introduction

Our group has decided upon developing the 64kb high-speed cache (memory option 2) as we believed speed to be an important metric. However, this is not to say that delay will be our only metric. We understand the growing importance of energy usage and area within VLSI design. As such, we propose to use a weighted metric of Active_Energy_per_Access×Delay^2×Area×Idle_Power. Deciding to use this metric allows us to emphasize delay while still keeping the other metrics in consideration.

In preparation of this design project, we created a proof-of-concept design review to demonstrate the functionality of our proposed SRAM bit cell. To demonstrate said functionality, we utilized the Cadence simulation software to create and test the SRAM while displaying simulation results. This data we found demonstrates that our SRAM needed minor adjustments before it was fully functional. Our pull-up and cell ratios needed to be refined to ensure proper latching of data in the SRAM cell.

Since then, we have modified our proposed design to address the functionality concerns. The new cell ratios are chosen as they are because it ensures proper writing and reading of data. The cell ratio was designed to make sure that the cell had the ability to properly read the values stored in Q and QB. The pull-up ratio was designed to ensure our cell was able to properly write values to Q and QB. Our original sizing was just upscaled, which was most likely a terrible idea that would have harmed our area metric. Back then, we were stressing only functionality. Now however, we need to begin to optimize our design. As such, we have reduced the size of our transistors to more reasonable sizes. After these optimizations and changes, our SRAM cell now functions as desired. See Figure 1 below for the updated (most importantly, working!) timing diagram.

The simulation in the diagram was a scenario that tested the important input combinations that could be desired. The scenario used was to write a logical 1 to the SRAM cell, read that 1, write a logical 0 to the cell, and then read that 0. All four stages of the given scenario appear to work fine in the transient plots. We've also updated our proposed layout attached as figure 2 (original as figure 3), but are still working further to refine the layout and guarantee a quality design.

With the new results, we are now confident that our SRAM is functional. Now, our next tasks involve optimization and improving the aforementioned metrics to ensure a superior design for PICo, with an emphasis on lowering delay times.

Proposed Approach/Architecture

There are many important considerations that must be accounted for in our design of a high-speed cache. Increasing the VDD of the circuit generally reduces the delay, but this action would result in a higher power dissipation that may or may not be an acceptable trade-off. Simulation results will be needed to verify the fruitfulness of increasing VDD for the sake of delay. Reducing the sizes of our transistors can reduce the effective delay by lowering the output capacitance. However there are other considerations we must take while modifying the sizes of the transistors. As mentioned earlier, it is vital to maintain the optimal cell ratios and pull-up ratios within each of the SRAM bit cells. Again, simulation will be needed to verify the optimal ratios beyond simple hand calculations.

Another way to reduce delay is to reduce the effective capacitive load experienced by the circuit. We have learned that splitting up the SRAM into smaller blocks reduces the delay brought about by capacitances from interconnects. As such, we are currently utilizing 32 smaller blocks to comprise the full SRAM block of 64kb. Each of the smaller blocks will have 32 columns (bit line pairs) and 64 rows (wordlines). This is equivalent to 2kb per block. The design will be further refined when we utilize simulation to find the best number of blocks (as well as rows and columns per block) with respect to our metrics.

Another method we will use to optimize our metric is dual V_T. This is the use of HVT and LVT MOSFETs in an appropriate combination. HVT transistors are slow but leak far less, saving power dissipation when able whereas LVT transistors are much faster and leaky. As we are more concerned with minimizing delay, we can utilize LVT transistors in our critical paths. However, changing every transistor to LVT would be very power consuming. We will have to find just how much of a tradeoff we are willing to make to ensure the best metric.

The sense amplifier we are currently utilizing is a differential voltage pair sense amplifier. We chose this particular sense amplifier because it capitalized on fewer transistors, resulting in a smaller area metric. Also, the sense amplifier uses the "READ" signal to turn on, which helps the sense amplifier dissipate less energy when not being used (refer to figure 4 attached below for a schematic of the sense amplifier). We will also research other sense amplifiers they may fit our metric better, such as the current sense amplifier.

As far as the architecture goes, we are using a fairly straightforward design. We will have the aforementioned 32 blocks of SRAM (comprising the full 64kb altogether) lined up. Then, we will utilize a hierarchical method of decoding to send the data used for writes where it needs to go. First, the data will enter a block demux which decides which block the data will go to. Then, it decodes even further into which columns within the block that the data must go to. Part of the address will be used for the decoding here. We chose this hierarchical structure for several reasons: (1) implementation is easier and quicker, (2) a hierarchical decoding requires less area, and (3) this will limit how much effective

capacitances the bit lines experience.

It is also necessary to act upon the specific rows. As such, we propose to use a decoder in tandem with the remaining bits of the address to specify the desired rows. Again, we will be using the hierarchical structure for the decoder to keep capacitances lower. Lastly, the output data during reads will go through the sense amp and another demux. Refer to figure 5 attached below for a block diagram of our proposed SRAM.

Of these described components of the block diagram, we will optimize each piece on its own. Through this methodology, we hope to create a highly optimized design in PICo's best interests. The first item optimization we will address is the decoders. Much like the SRAM blocks, the decoders and can become quite large and the long interconnects may result in excessive capacitance. We will need further research into different typologies of decoders that will meet and emphasize our metric of speed. With regards to speed and minimal delay, the transmission gate decoders are generally both faster and smaller in area so we will have to test and simulate that as a typology. The same can be said about the MUXes that we will need to implement. The transmission gates can be a quick and area-conscious way to implement them.

Novelty of our Design

The efficiency of the SRAM is not only determined by the actual 6T cell, but the other components that enter into the overall design. Our idea is if each one of our peripheral components is optimized for speed, then the overall SRAM will thereby be optimized for speed. To find novelty ideas and designs we will concentrate on advanced SRAM techniques learned in class - e.g. power-gating to improve power-consumption. For innovative new ideas, we will do research and test designs from US Patents and IEEE's publication repository - e.g. "US Patent No. 5991217: Fast SRAM Design Using Embedded Sense Amps," "A current sense-amplifier for fast CMOS SRAMs," "Low-power fast static random access memory cell," "US Patent No. 7143257: Method and Apparatus of a Smart Decoding Scheme for Fast Synchronous Read in a Memory System," and "US Patent No. 5604712: Fast Word Line Decoder for Memory Devices." Lastly, we will want to use a fast precharge as well.

With regard to the actual SRAM bit cell, we will test various cell ratios and widths to see which provide the best output. Also, various designs exist for faster SRAM cells, which we will investigate (some of which were mentioned in the previous section regarding patents). Lastly, we will conduct thorough Monte Carlo simulations as well as ensure the integrity of our signal throughout this whole process because speed is only useful when the data is correct.

Closing Remarks

As time is limited and our goal is to make the best circuit yet in respect to our chosen metric, we will utilize much of the previous years work to help supplement our work and speed up the implementation of our design. Using SKILL and OCEAN coding and simulation will help us alleviate some of the time constraints and hopefully leave us additional time for further optimization or implementing additional novelties that will make our design superior to any prior designs.

Attached below are updated simulations demonstrating the functionality of our SRAM bit cell, relevant updated schematics, a project tasks breakdown, and a proposed timeline showing when we hope to have certain parts of the project completed.

Project Tasks Breakdown

Robert Costanzo

- Design & Simulation
    - Decoder
- Architecture
- Overall simulation results
- High Level Layout
- Optimizations for interconnects and decoder

Michael Recachinas

- Design & Simulation
    - Sense Amplifier
- Schematics
- Optimizations for Sense Amplifier and 6T Bitcell
- Project Journal
- Updating Wiki

Hector Soto

- Design & Simulation
    - 6T Bitcell
- Clock & Timing
- Process Corner Simulation
- Optimizations for Word Lines & Bit Lines

Project Timeline and Milestones (Weekly)

Note: bolded milestones on schedule are customer deadlines (i.e. Design Reviews, Presentations)

- Oct 13 - 19
    - Oct 16: Optimize layout before beginning speed optimizations
    - Oct 16: Ensure all simulations show functional SRAM cell
    - Oct 16: Finish proposal for PICo
    - **Oct 17: Submit proposal to PICo**
- Oct 20 - 26
    - Oct 20: Begin in depth learning of SKILL and writing Ocean scripts
    - Oct 21: Connect all components together and simulate
    - Oct 22: Calculate initial pre-optimization metrics

    - Oct 24: Begin designing schematics for high speed optimizations
- Oct 27 - Nov 2
    - Oct 28: Conduct simulations on optimized schematics
    - Oct 29: Begin connecting components together and testing
- Nov 3 - 9
    - Nov 3: Ensure all components are connected and tests show optimized functionality
    - Nov 4: Finish layout for optimized SRAM
    - Nov 9: Finish process corner simulations for optimized SRAM
- Nov 10 - 16
    - Nov 11: Ensure all simulations show optimized functionality for SRAM
    - Nov 11: Finish report for Design Review 2
    - **Nov 12: Submit Design Review 2**
- Nov 17 - 23
    - Nov 17: Finish Design and basic simulations
    - Nov 21: Finish process corner simulations
    - Nov 22: Calculate design metrics and process corner metrics
    - Nov 23: Finish layout for entire design
- Nov 24 - 30
    - Nov 26: Ensure all simulations are accurate and represent our design well
    - Nov 28: Write final design report and presentation for PICo
- Dec 1 - 5
    - **Dec 3: Ensure everything is finalized and prepared for delivery**
    - **Dec 3 or 5: Present Design to PICo**

Wiki

Our website is located at:
http://venividiwiki.ee.virginia.edu/mediawiki/index.php/ClassECE4332Fall13GroupVeryLargeScaleEngineers

Appendix (Figures)

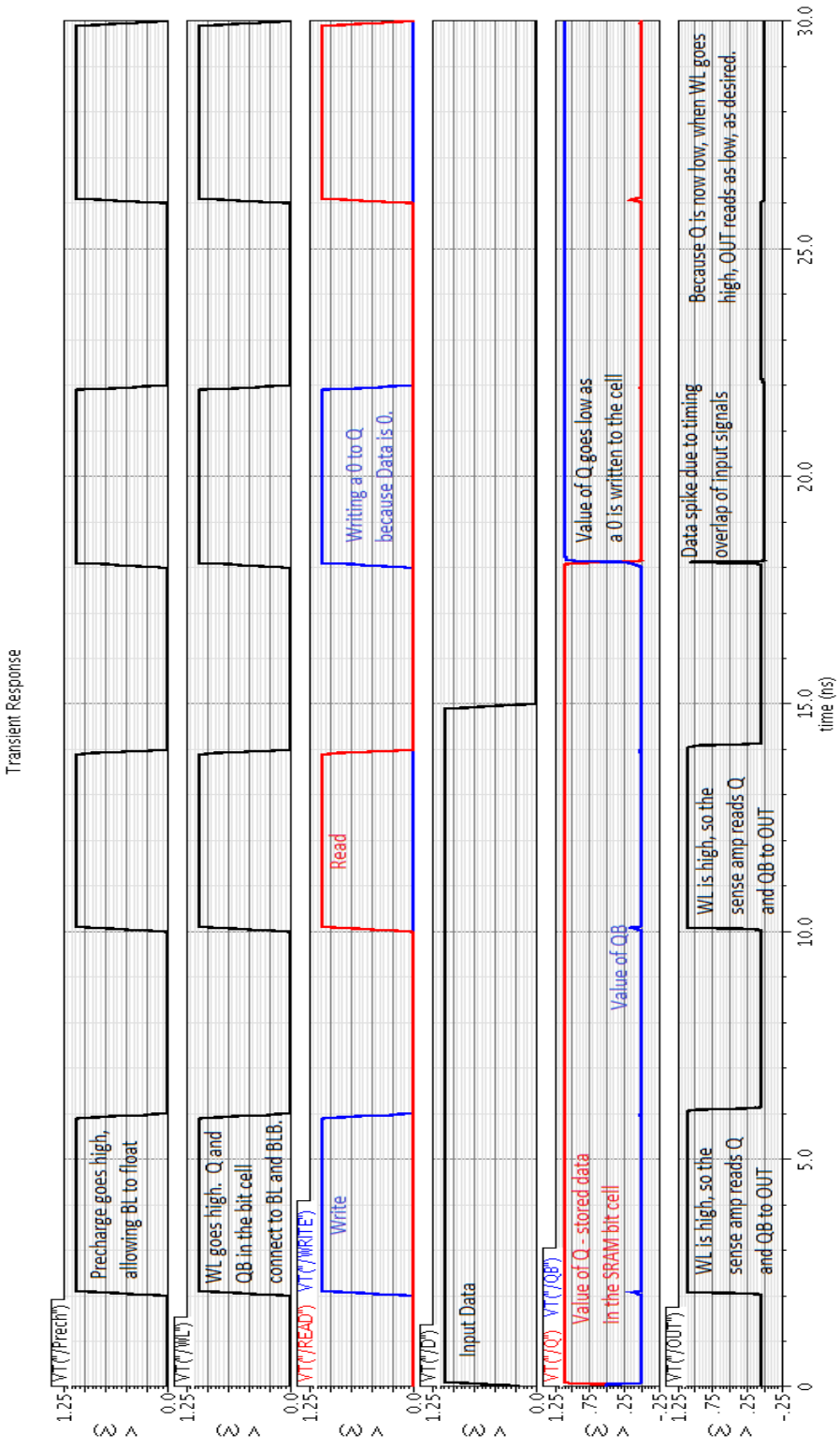

Figure 1 Timing Diagram of Simulation Tests of SRAM Bit Cell

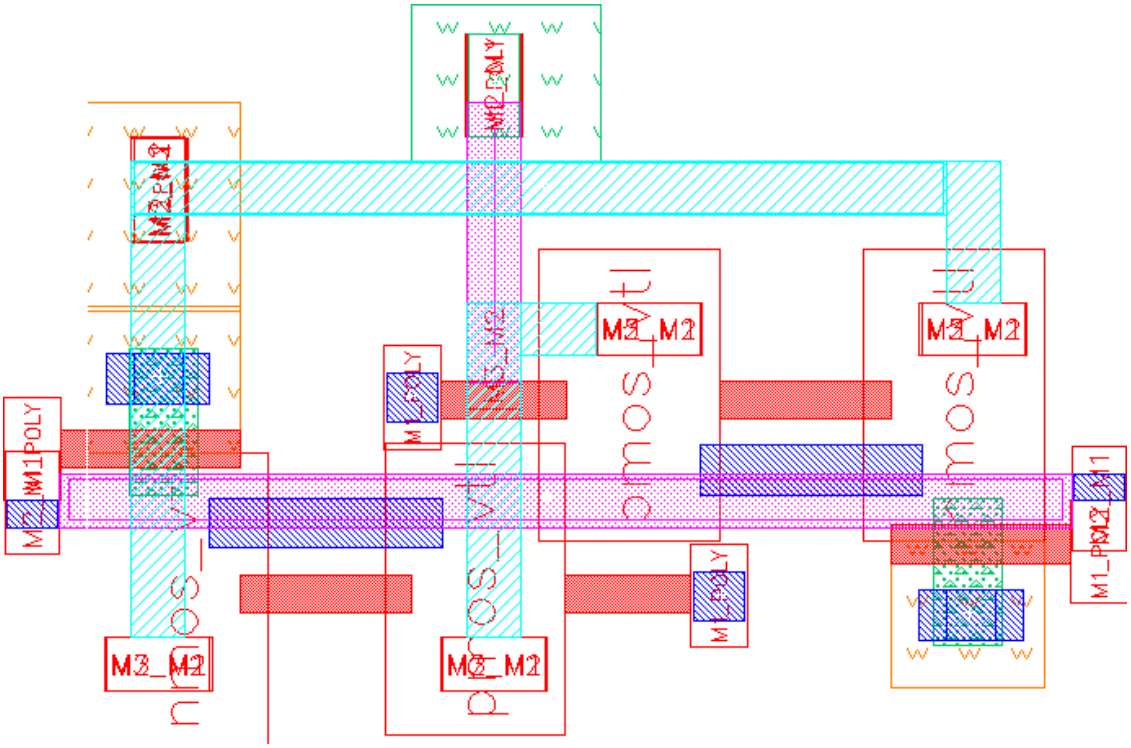

Figure 2 Proposed Change to Layout

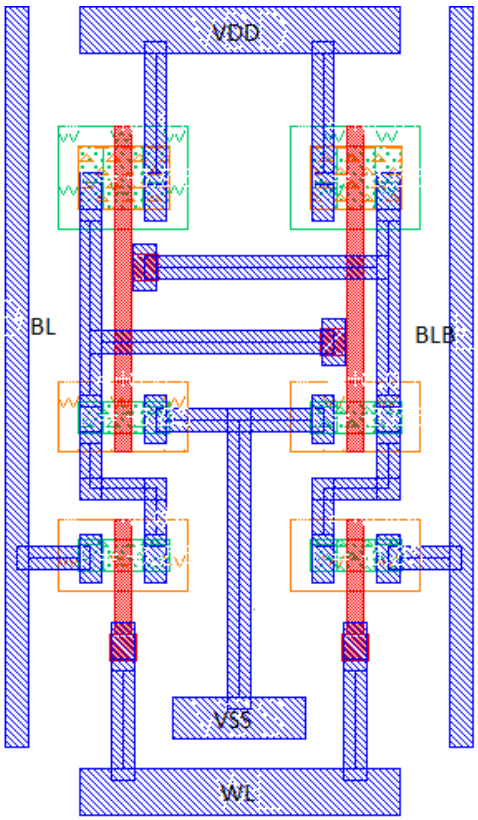

Figure 3 Original Layout (for comparison)

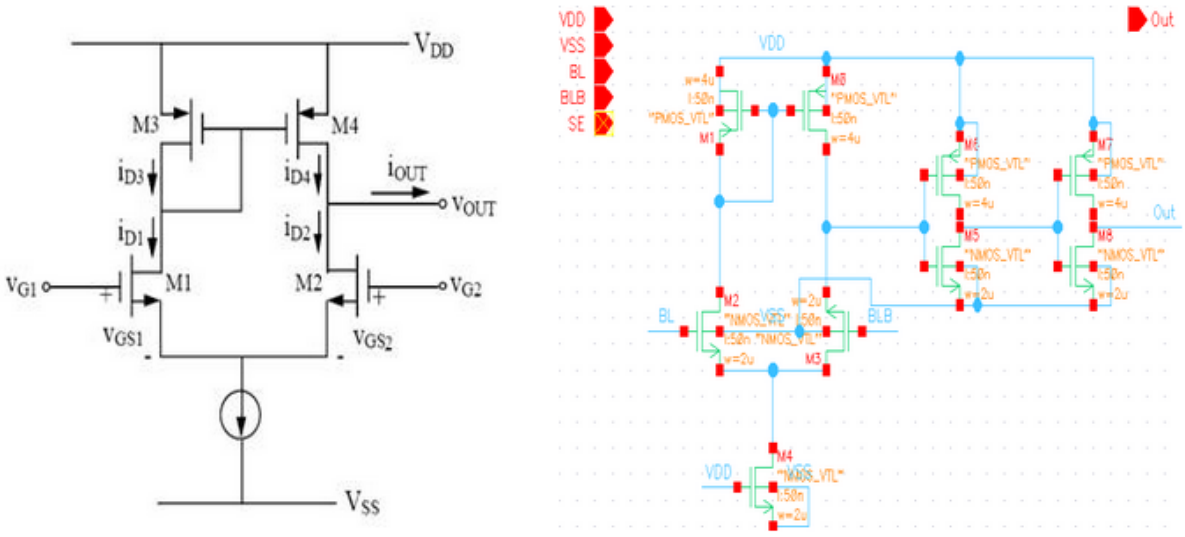

Figure 4 Sense Amplifier Schematic - Traditional Design[5], left; Our design, right

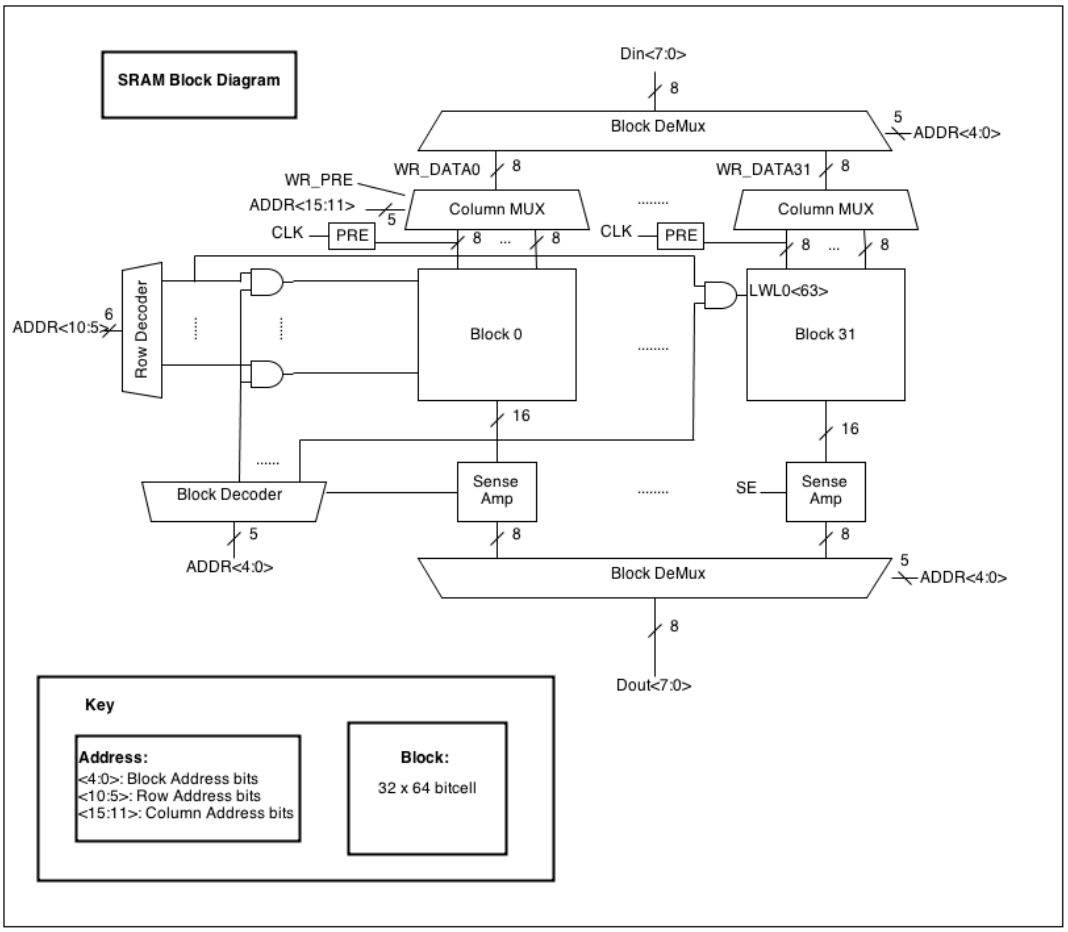

Figure 5 Block Diagram of Proposed SRAM (modified diagram from previous group[3])

References

1. Rabaey, J., Chandrakasan A., Nikolic, B., *Digital Integrated Circuits (2nd Edition)*, (Dec 24, 2002)
2. Calhoun, B., *Design Principles for Digital CMOS Integrated Circuits*, (March 7, 2012)
3. Cash L., Duan C., Reed R., Tyler A., *Design Review 1, Los tOHMales Callentes, VLSI 4332*, (Oct 5, 2012)
4. Bailey S., Linger K., Lorenzo R., Thompson J., *Project SRAM - Design Review 1*, (Oct 2012)
5. *Design And Analysis Of Sense Amplifier*, http://www.projecttopics.info/Computer/Design_And_Analysis_Of_Sense_Amplifier.php
6. Jacob, B., Ng, S. W., & Wang, D. T. *Memory systems: cache, DRAM, disk,* Burlington: Morgan Kaufmann, 2007, 282.
7. Amrutur, B., *Design and Analysis of Fast Low Power SRAMs*, (Aug 1999)
8. Prabhu, C.M.R., *Low-power fast static random access memory cell*, (Dec 2010)
9. US 7499312: "Fast, stable, SRAM cell using seven devices and hierarchical bit/sense line"
10. US 5604712: "Fast Word Line Decoder for Memory Devices"
11. US 5991217: "Fast SRAM Design Using Embedded Sense Amps"
12. US 7143257: "Method and Apparatus of a Smart Decoding Scheme for Fast Synchronous Read in a Memory System"